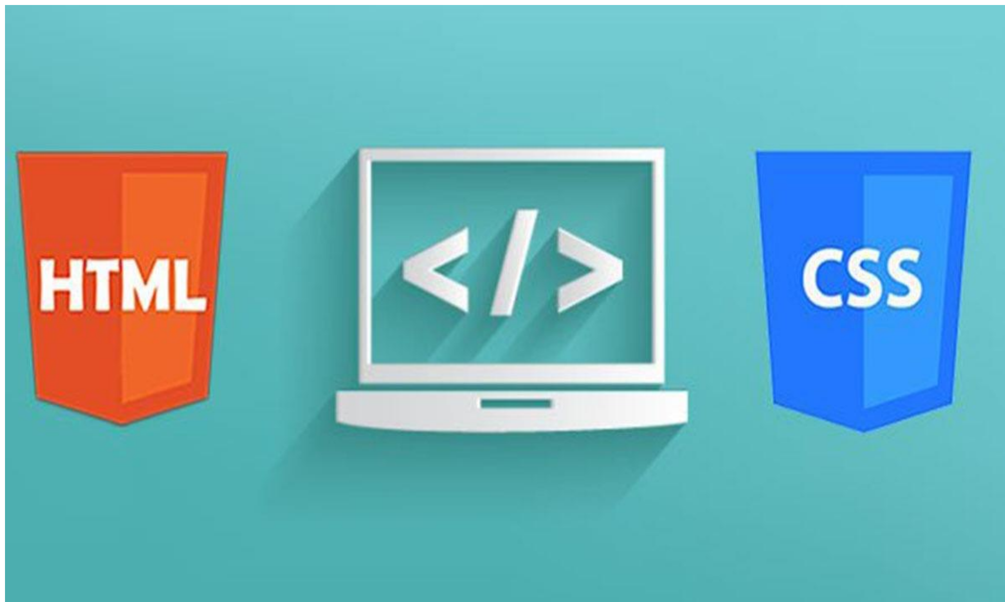
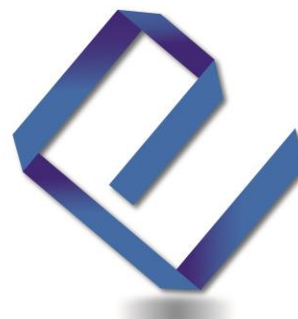




Preparation:
Mohammed Alhakem
Tutor



Formatting:
ESRAA ELSHIEKH

Content is Lessons from the one Million Arab Coders initiative

to contact us: info@alhakem.me

حتى تبدأ في تعلم أي لغة برمجية جديدة يجب عليك أولاً أن تعرف لماذا تحتاج هذه اللغة وما وظيفتها حتى تستطيع البدء ولديك رؤية وأضحة عن ماذا ستستطيع القيام به بإستخدامها مما قد يسهل عليك عملية دراستها.

لغة HTML

هي اختصار لـ Hyper Text Markup Language وفي الحقيقة لا تصنف كلغة برمجة بل كلغة توصيف فهي تقوم بعمل توصيف لمحتويات الموقع ولا تستطيع القيام بعمل وظيفي كالقيام بالعمليات الحسابية.

تعتبر HTML وحدة البناء الأساسية للموقع (هيكل الموقع)



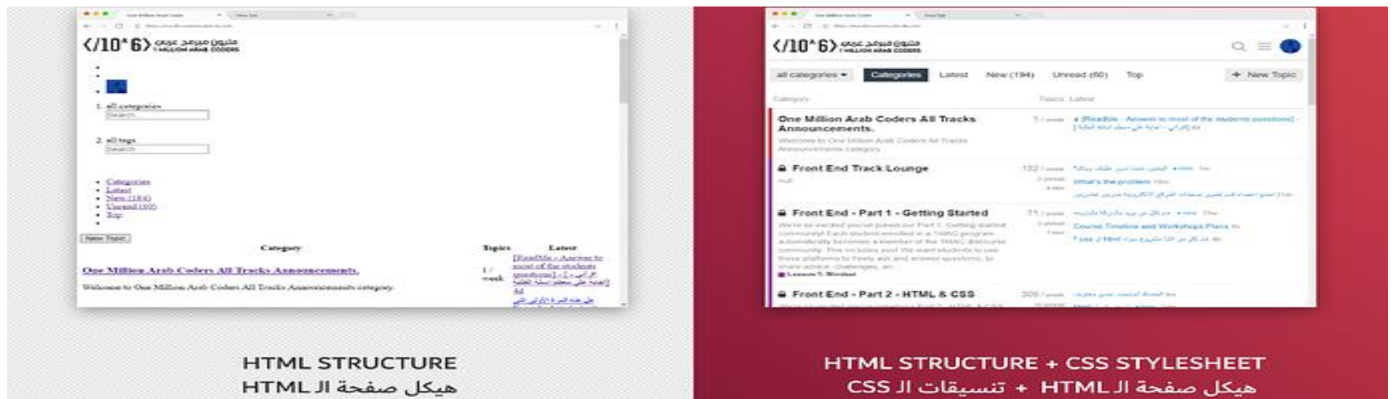
لغة CSS

هي اختصار لـ Cascading Style Sheet و هي لغة تصميم المواقع والمعنية بإضافة التنسيقات لهيكل الصفحة من حيث لون الخلفية او نوع الخط على سبيل المثال.



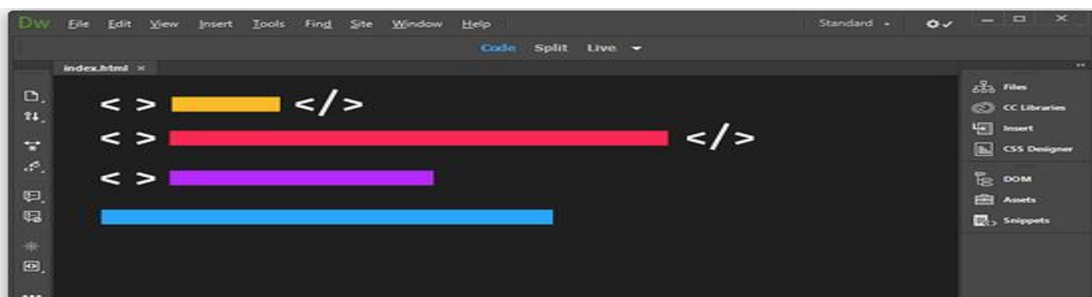
إذاً نستطيع أن نقول بأن لغة HTML هي المسؤولة عن هيكلة و وضع محتوى الموقع بينما CSS هي اللغة المسؤولة عن تصميم وتنسيق هذه المحتويات.

لنأخذ كمثال منتدى المبادرة



نلاحظ كيف أن HTML تهتم بهيكلة الصفحة و وضع المحتوى بينما CSS تهتم بتصميم الصفحة بترتيب العناصر و إضافة التنسيقات لها.

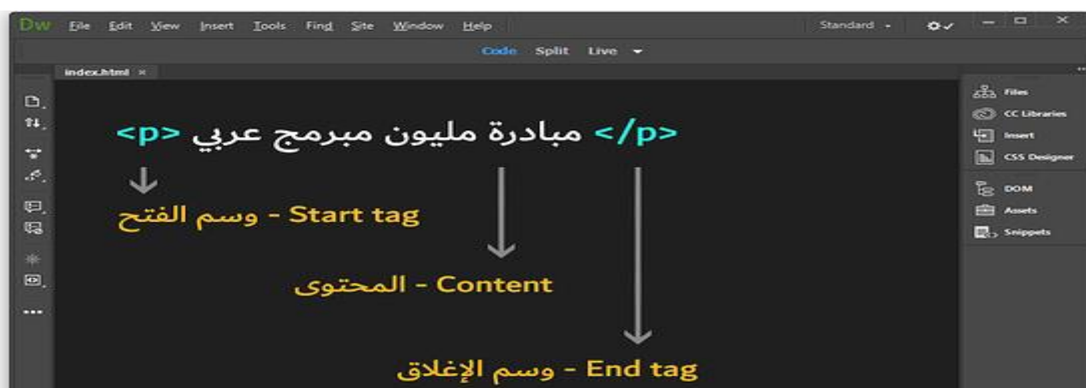
تعتبر الوسوم (Tags) هي البنية الأساسية لأي مستند HTML حيث أنها هي المسؤولة عن تحديد نوع عناصر الصفحة من صور، فقرات، عناوين، جداول وغيرها.



هناك نوعين من وسوم الـ HTML

ـ وسوم تحتاج الى فتح و إغلاق:

هذا النوع من الوسوم يكون لديه محتوى محدد للتأثير عليه لهذا يجب عليك أن تقوم بتحويط النصوص أو العناصر المراد إستهدافها بوسم معين حتى لا تتداخل الوسوم في بعضها البعض.



على سبيل المثال وسم

هذا وسم يقوم بالتأكيد على نص معين وكتابته بالخط العريض. عندما تريد إستهداف نص معين بهذا الوسم يجب عليك كتابة هذا النص بين وسمَي الفتح والإغلاق كالتالي

```
One million <strong>Arab</strong> Coder
```

نتيجة الكود كالتالي

One million **Arab** Coder

لكن لنفترض بأنك نسيت كتابة وسم الإغلاق كالتالي :

```
One million <strong> Arab Coder
```

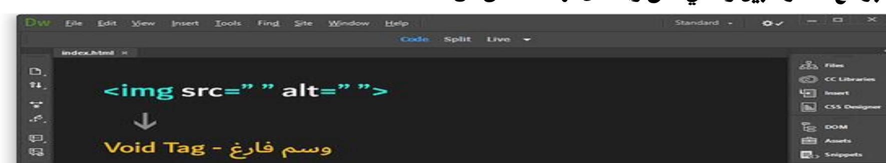
النتيجة ستكون كالتالي:

One million **Arab Coder**

لاحظ كيف تم التأكيد على جميع النصوص بعد وسم الفتح وكتابتها بالخط العريض وذلك لأن مستند الـ HTML **لم يقوم بتعريف حدود تأثير هذا الوسم بإستخدام وسم الإغلاق.**

ـ وسوم لا تحتاج إلى وسم إغلاق:

هذه الوسوم لا تحتاج إلى أن تقوم بوضع محتوى بين وسمَي فتح وإغلاق لهذا تسمى Void tags



على سبيل المثال وسم <img> **كل ما تحتاج إلية لإضافة صورة هو تحديد مكانها وكتابة النص البديل الذي سيتم عرضة في حال لم يتم العثور على الصورة بالمسار الذي قمت بتحديدة.**

```
<img src="img.jpg" alt="my image">
```

لا تحتاج أن تقوم بكتابة الوسم كالتالي:

```
<img src=" " alt=" ">هنا محتوى</img>
```

حيث أنه لا يقوم بإستهداف محتوى محدد إنما يقوم بإدراج صورة.

مستند الـ HTML **يتكون من العديد من العناصر التي نعتبرها كوحدات البناء والهيكلة لصفحة الويب وحتى يتم عمل التنسيقات لهذه العناصر وتحديد طريقة عرضها تم تصنيفها إلى نوعين.**



عناصر نوع Block :

هذه العناصر بإمكانك تخيلها كصندوق عند وضعة على صفحتك فسوف يأخذ مساحة العرض كاملة مما يعني أنه لا يقبل بأن يشاركة أي عنصر أخر بنفس السطر.

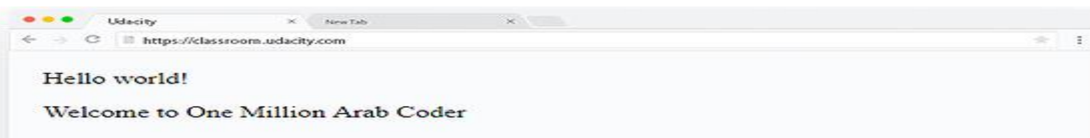


أمثلة:

لنقوم بفتح محرر النصوص وكتابة الكود التالي:

```
<p>Hello world!</p> <p>Welcome to One Million Arab Coder</p>
```

النتيجة:



كما تلاحظ تم كتابة كل عنصر في سطر منفصل مع اننا لم نحدد أننا نريد ذلك بشكل او بأخر

عناصر نوع Inline:

بعكس العناصر من نوع Block هذه العناصر تشارك بعضها المساحة ولا تأخذ العرض كامل وبإمكانك إدراج أكثر من عنصر بجوار بعضها البعض.



أمثلة:

لنقوم بفتح محرر النصوص وكتابة الكود التالي:

```
<span>Hello World! </span>
<span>Welcome to One Million Arab Coder</span>
```

النتيجة:



لاحظ كيف تم عرض العناصر هذه المرة في سطر واحد

في الختام:

- لا يجب عليك أن تحفظ أي العناصر هي block وايها inline بل أن تفهم الفكرة ومع الوقت و الممارسة ستستطيع تحديد ذلك كونك قد سبق وأستخدمت العنصر الفلاني وتعرف طريقة عرضة وعملة.
- يوجد في المسار كويز لهذا الموضوع واعتقد أن الأغلب قد مروا به، ما اتمناه هو انكم قد أستطعتم معرفة الجواب عبر تطبيق ذلك بكتابة الأكواد في محرر النصوص وتحديد النوع من خلال مشاهدة النتيجة كما قمنا في الأمثلة أعلاة.
- ربما قد تكون تتسأل عن نوع وسم
 و وسم <hr> قم بالتجربة والبحث عن الإجابة واطلعنا عليها

عناصر div, p and span والفرق بينهم

في الدرس السابق تكلمنا عن عناصر الـ block و الـ inline وفي هذا الدرس سنتكلم عن بعض أهم العناصر التي سنتحتاج أن نفهمها قبل ان نبدأ في التطبيق على أي مشروع.

وسم Div

المرة القادمة عندما تشاهد اي موقع قم بتخيل وجود صناديق وهمية تضم عناصرة كما بالصورة التالية:



نستخدم وسم <div> لتقسيم الموقع لاجزء معينة حتى نستطيع تطبيق تصميم معين عليها باستخدام لغة التنسيقات CSS

- جميع العناصر يمكن ضمها بداخل العنصر div

```
<div>
```

```
  <!--other elements-->
</div>
```

مثال :

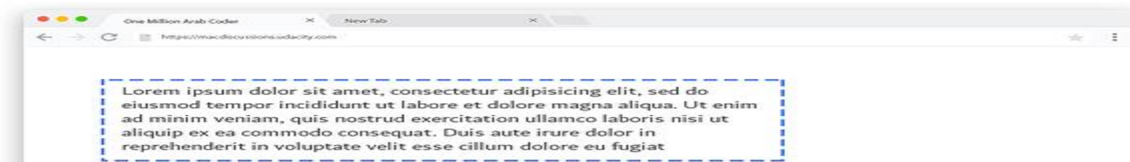
```
<div>
   <h1>hello world!</h1>
</div>
```

النتيجة:

hello world!

وسم p

كذلك في المرة القادمة عندما تشاهد فقرة ما في أي موقع قم بتخيل صندوق وهمي حولها ولكنة خاص بالنصوص فقط هذه المرة



حيث نستخدم وسمَ <p> لكتابة الفقرات (paragraph) فقط وتطبيق تصميم معين عليها بإستخدام لغة التنسيقات CSS

- لا يمكنك ضم أي عنصر بداخل هذا الوسم الا اذا كان عنصر لعمل تنسيقات على النصوص مثل وسم

```
<p>
<!--only text and text formating tags-->
   <!--هنا فقط نصوص و وسوم تنسيق النصوص-->
</p>
```

مثال :

```
<p>
  <strong>hello <em>world!</em></strong>
</p>
```

النتيجة:

hello *world!*

لا يمكنك وضع div بداخل p ولكن العكس صحيح.

```
<p>
   <div>هذا الكود غير صحيح</div>
</p>
```

وسم span

هذا الوسم كذلك خاص بالنصوص فيتم إستخدامة لعمل تنسيق معين لجزء معين من النص

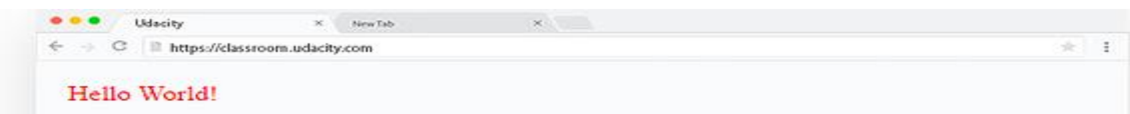


مثال:

قمنا بعمل فقرة بإستخدام وسم <p> وعملنا لها استايل يغير لون الفقرة الى الأحمر

```
<p style="color:red;">
     Hello World!
</p>
```

هذه النتيجة:



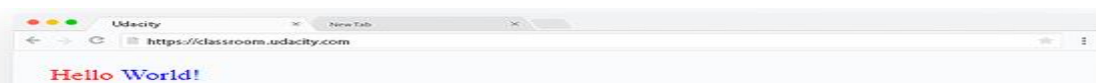
لنفترض أننا نريد أن نغير لون كلمة World ! وحدها للون الأزرق

في هذه الحالة سنستخدم وسم <span> لان وجودة لا يؤثر على الجملة الا اذا تم تحديد تنسيق معين له.

في الكود التالي تم تحويط كلمة World ! بإستخدام وسم span وتم إضافة تنسيق تغيير اللون للازرق

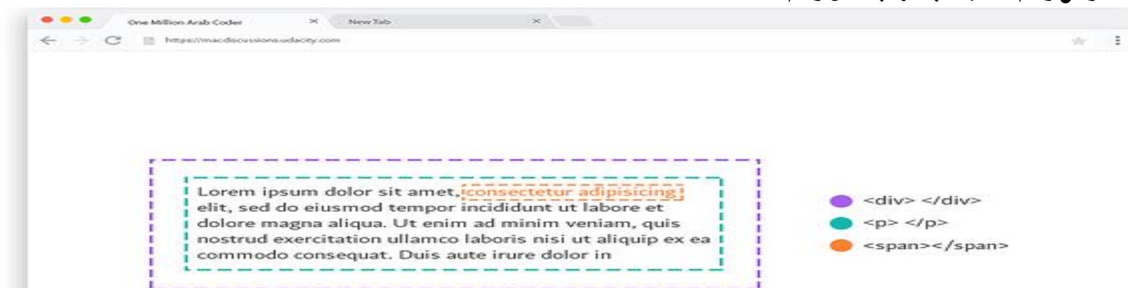
```
<p style="color:red;">
    Hello <span style="color:blue;">World!</span>
</p>
```

النتيجة:



في الختام تذكر التالي:

- دائماً وسمي الـ p و الـ span تأتي بداخل الـ div.
- لا تقم بوضع وسم div بداخل وسم p او span.
- بإمكانك وضع وسم span مباشرة بداخل وسم div.



حتى هذه اللحظة تعرفنا على عناصر الـ HTML وأنواع الوسوم و تبقى لنا أن نناقش سؤال مهم وهو:

كيف أستطيع الوصول لأي عنصر محدد في الصفحة والتلاعب به بإضافة التنسيقات عليه؟

قبل الجواب على هذا السؤال لنأخذ مثال بسيط عن عضويتك بهذه المبادرة



لنفترض أني أستطيع التحكم بالعضويات وأنت الأن كعنصر في هذه المبادرة يمكنني الوصول اليك بطريقتين:

الطريقة الأولى: عبر إيميلك الشخصي

- إذا ما أعطيتني إيميلك الشخصي فسأستطيع الوصول اليك أنت وحدك مباشرة والتأثير على عضويتك بطريقة أو بأخرى.
- تلاحظ أن أيميلك الشخصي لا يمكن لأحد أن يستخدمة أو يحصل على أيميل مطابق له فهو يخصك أنت وحدك.

الطريقة الثانية: عبر المجموعة التي تم ضمك لها

- أنت كعنصر في هذه المبادرة تم ضمك لمجموعة مسار تطوير واجهات الويب فاذا انا ما قمت بالتأثير والتعديل على هذه المجموعة فعضويتك ستتأثر كذلك.
- تلاحظ أن هذه المجموعة أنت تشاركها مع بقية الأعضاء وليست شئ خاص بك وحدك والتعديل عليها يؤثر عليكم جميعاً.

الطريقة الأولى كانت بإختصار ما يسمى بالـ id

حيث وانه يتم منح العنصر مسمى يميزة عن غيرة حتى نستطيع أن نستهدفة ونتلاعب بة دون غيره.

يجب أن تعرف عن الـ id التالي:

- لكل عنصر id واحد.
- تحتوي الصفحة الواحدة على عنصر وحيد يحمل Id فريد خاص به دون غيرة

مثال:

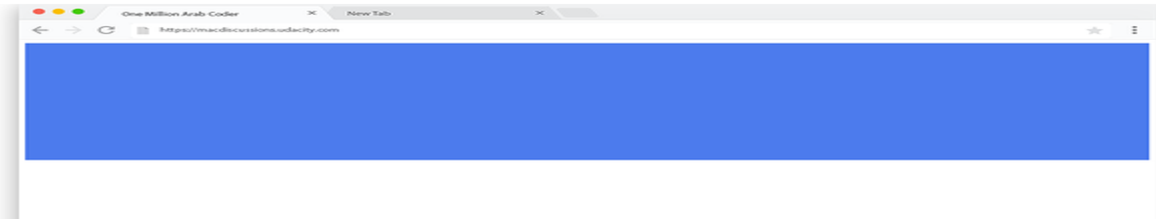
```
<div id="header">
<!--Header content here-->
</div>
```

- لاحظ قمت بإعطاء هذا الـ div هوية تعريفية خاصة به وحدة (id) بمعنى أنه لا يمكنني أن أقوم بإنشاء عنصر أخر وأعطية نفس الـ id.
- أستطيع الأن أن اقوم بإضافة تنسيقات لهذا العنصر دون سواة بإستخدام الـ CSS عن طريق إستهدافة بإسم الـ id الذي منحتة كالتالي:

```
<style>
    #header{
         width:100%;
         height:200px;
         background: #4c7bed;
      }
</style>
```

لاحظ عند تعريف الـ id في الـ CSS نقوم بوضع # قبل إسم الـ id

النتيجة:



الكود كاملاً

```
<!doctype html>
<html>
<head>
    <meta charset="utf-8">
    <title>One Million Arab Coder</title>
    <style>
    #header{
       width:100%;
       height:200px;
       background: #4c7bed;
      }
    </style>
</head>
<body>
<div id="header">
<!--Header content here-->
</div>
</body>
</html>
```

الطريقة الثانية كانت بإختصار ما يسمى بالـ class

حيث وانه يتم منح العنصر او مجموعة من العناصر مسمى يميزهم عن غيرهم حتى نستطيع أن نستهدفهم ونتلاعب بهم دون غيرهم .

يجب أن تعرف عن الـ class التالي:

- يمكن أن يُستخدم الـ class لأكثر من عنصر.
- يمكن أن يُستخدم أكثر من class للعنصر الواحد.

مثال:

```
<p class="paragraph">
    Hello world!
</p>
<p class="paragraph">
    One Million Arab Coder
</p>
```

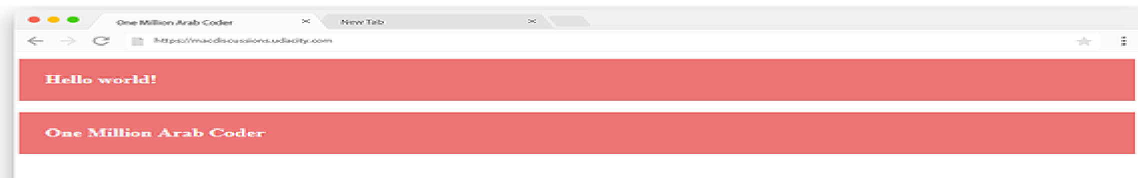
- لاحظ قمت بعمل عنصرين وإعطائهما نفس الـ class حيث اني الان أستطيع أن اضيف التنسيق مرة واحدة وسيتم تطبيقة على جميع العناصر التي تحمل نفس أسم الـ class

مثال:

```
<style>
   .paragraph{
      color:#fff;
      background: #ed7474;
      padding: 20px;
      font-weight: bold;
     }
</style>
```

لاحظ عند تعريف الـ class في الـ CSS نقوم بوضع . قبل إسم الـ class

النتيجة:

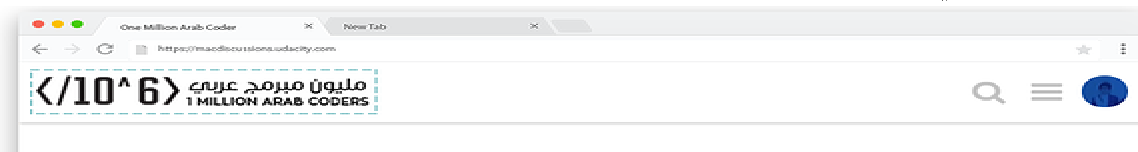


الكود كاملا

```
<!doctype html>
<html>
<head>
     <meta charset="utf-8">
     <title>One Million Arab Coder</title>
     <style>
    .paragraph{
       color:#fff;
       background: #ed7474;
       padding: 20px;
       font-weight: bold;
     }
</style>
</head>
<body>
<p class="paragraph">
    Hello world!
</p>
<p class="paragraph">
    One Million Arab Coder
</p>
</body>
</html>
```

كيف أعرف متى أستخدم الـ `id` او الـ `class` ؟

الـ id تقوم بإستخدامة مع العناصر التي لا تتكرر في الصفحة لنأخذ على سبيل المثال الشعار في هذا المنتدى بإمكاننا إعطائة id



اما الـ class تقوم بإستخدامة مع العناصر التي تتكرر في الصفحة لنأخذ على سبيل المثال هذا الجزء في المنتدى حيث يتكرر نفس التنسيق.

لكن هذا لا يعني أنه لا يمكنك إعطاء الشعار class بدلاً من id فالأمر راجع إليك.

هل يمكنني إعطاء العنصر Class و id بنفس الوقت؟

أرجع لمثالنا عن عضويتك بهذه المبادرة هل أنت تملك id و class بنفس الوقت ؟

اذا فالوضع ليس مختلف عن هذا

في الختام أحرص على أن تفهم الفكرة الرئيسية ولا تقلق من أكواد التنسيقات حيث أننا سنفصلها لاحقاً وستصبح مألوفة لديك مع الممارسة

الهيكل الأساسي لصفحة الويب

في هذا الدرس سنتعرف على البنية الأساسية لأي مستند HTML

```
<!doctype html>
<html>
<head>
    <meta charset="utf-8">
    <title>Untitled Document</title>
</head>
<body>
</body>
</html>
```

- قم في بداية الكود بتحديد نوع مستند الـ HTML الذي سنعمل علية في هذه الحالة سنستخدم HTML5 ولتحديد ذلك نستخدم الكود التالي:

```
<!doctype html>
```

لا تقوم بوضع أي كود قبل كود تعريف نوع المستند حتى وان كان تعليق

- **منطقة الـ head تستخدم بشكل خاص لـ metadata (البيانات الوصفية)**
- **كعنوان الصفحة،**
- **ترميز الصفحة،**
- **وصف الصفحة،**
- **الكلمات المفتاحية وغيرها من البيانات التي تكون مفيدة لعرض بيانات صفحتك عند البحث عليها على قوقل على سبيل المثال.**
- **كذلك سنقوم بتعريف التنسيقات او اكواد الجافاسكربت في هذه المنطقة اما بشكل مباشر او عن طريق وضع وسوم تقوم بربط الصفحة بملفات خارجية.**

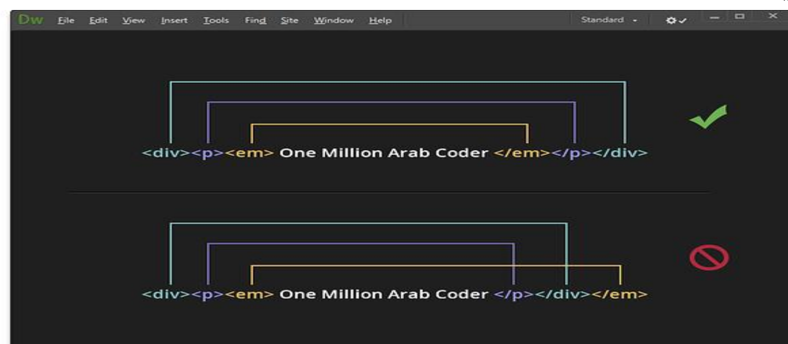
منطقة الـ body: هي منطقة جسم الصفحة والتي سنضع فيها كل العناصر من صور ونصوص وجداول وغيرها.



وسم	`<meta charset="utf-8">`

ليس من العناصر الأساسية ولكن حتى نتجنب أي مشاكل مع ترميز اللغة العربية قم بوضعة دائماً قبل أي شي حتى قبل وسم الـ Title.

- **ملاحظة هامة: عند قيامك بكتابة أكواد الـ HTML فكر كأنك تصنع ساندويتش 🥪 وتأكد من فتح و إغلاق كل وسم في المكان المناسب شاهد الصورة التالية حتى تفهم قصدي:**



في هذه المرحلة يفترض أن الجميع قد أصبح قادر على إنشاء صفحة ويب وكتابة أكواد البنية الأساسية ويعرفها جيداً لهذا لن أتكلم بأي تفاصيل مملة فقط أحببت على أن نؤكد بأن هذه هي أهم الوسوم التي الأساسية والتي ستكون حجر الأساس لمشاريعنا القادمة فاذا لديك أي إستفسار ارجوا وضعة بالتعليقات.

طرق إستخدام أكواد CSS في صفحات HTML

سنشرح ثلاث طرق ونجرب عليها أمثلة تطبيقية



الطريقة الأولى: أضافة أكواد CSS لوسم HTML مباشرة

مثال:

```
<!doctype html>
<html>
<head>
    <title>One Million Arab Coders</title>
</head>
<body style="background-color: black;">
    <p style="color:white; font-size:25px;">One million Arab Coders</p>
</body>
</html>
```

النتيجة:



الذي قمت بعملة هو أضافة خاصية style **لوسم** body **وتغيير لون الخلفية للأسود بإستخدام خاصية** background-color **كما في التالي**

```
<body style="background-color: black;">
```

واضفت style **لوسم** p **وقمت بتغيير لون الخط وحجمة**

```
<p style="color:white; font-size:25px;">One million Arab Coders</p>
```

لا ينصح بإستخدام هذه الطريقة لإضافة أكواد ال CSS **لعناصر** HTML **وذلك لانه عند كتابتك للملف** HTML **لنقول يتكون من ١٠٠٠ سطر وقتها إضافة ال** CSS **بهذه الطريقة سيزيد حجم الملف و سيكون من الصعب عمل التعديلات على التصميم ناهيك عن التقسيم والترتيب السئ للاكواد.**

الطريقة الثانية: تضمين أكواد ال CSS بداخل مستند HTML بشكل مباشر

هذه الطريقة أنظف من السابقة حيث بإمكانك كتابة أكواد ال CSS **بداخل وسم** style

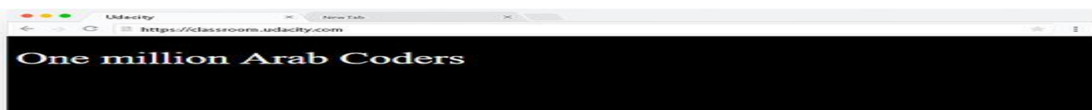
```
<style>هنا الأكواد</style>
```

يتم وضع هذا الوسم في منطقة الhead

كالتالي:

```
<!doctype html>
<html>
<head>
    <title>One Million Arab Coders</title>
    <style>
      body{
            background-color:black;
            }
            .paragraph{
              color:white;
              font-size:25px;
            }
    </style>
</head>
<body>
    <p class="paragraph">One million Arab Coders</p>
</body>
</html>
```

النتيجة كما في التصميم السابق:



في هذا الكود تم استهداف وسم body **وتغيير لون خلفيته**

```
body{
  background-color:black; /*تغيير لون الخلفية للأسود*/
}
```

وبنفس الفكرة مع وسم p قمنا بإعطائة إسم كلاس paragraph وإستخدمناه لإضافة التنسيقات كالتالي:

```
.paragraph{
  color:white; /*تغيير لون الخط للأبيض*/
  font-size:25px; /*تغيير حجم الخط*/
}
```

ما يميز هذه الطريقة عن التي قبلها أنك تقوم بكتابة الأكواد في منطقة واحدة مما يسهل عليك عمل التعديلات لكن يضل لها تكلفة في حال كان حجم الكود كبير

الطريقة الثالثة: ربط مستند HTML بملف CSS منفصل

في هذه الطريقة تقوم بإنشاء ملفين منفصلين الأول خاص بأكواد ال HTML والثاني خاص بأكواد ال CSS ثم تقوم بربط ملف ال HTML بملف التنسيقات عن طريق إستيرادة بإستخدام كود HTML التالي:

```
<link rel="stylesheet" type="text/css" href="style.css" />
```

حيث تم فية تحديد موقع واسم الملف الخاص بال CSS والذي أسمة style.css ويتم وضع هذا الكود في منطقة ال head كالتالي:

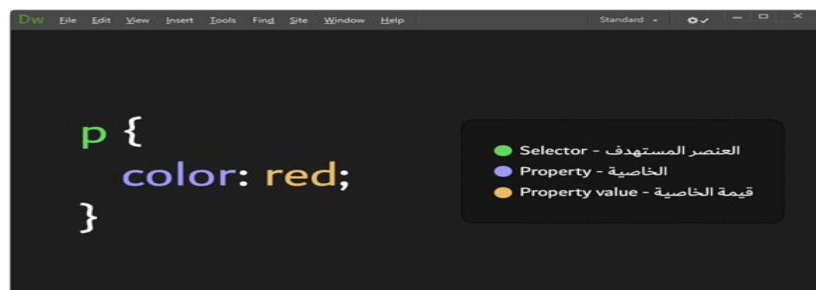
```
<!doctype html>
<html>
<head>
   <title>One Million Arab Coders</title>
   <link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>
   <p>One million Arab Coders</p>
</body>
</html>
```

الان بداخل ملف style.css سأقوم بتحديد الخصائص كما قمنا في الطريقة الثانية ولكن هذه المرة لا نحتاج لتحويطها بوسم style كونه قد تم تحديد نوع الملف بأنه ملف تنسيقات CSS:

```
body{
   background-color:black;
}
p{
   color:white;
   font-size:25px;
}
```

هذه تعتبر أفضل طريقة لكتابة أكواد HTML حيث أن فصل ملف التنسيقات عن صفحة ال HTML يساعدنا على ربطها بأكثر من صفحة وتطبيق نفس التنسيقات على الصفحات المرتبطة بها وكذلك عمل التعديل على التنسيقات بملف واحد وتطبيقة على أكثر من ملف HTML بالإضافة إلى ان فصل الملفات بهذه الطريقة يساعد على تنظيم الأكواد وتقسيم ملفات الموقع.

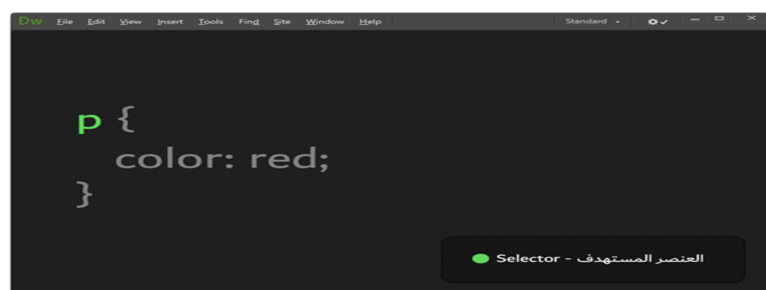
في هذا الدرس سنتكلم عن أهم الأساسيات لكتابة أكواد الـCSS.



الكود في الصورة أعلاة هو مثال لأبسط المكونات الأساسية لعمل تنسيق CSS لعناصر الـ HTML وسنفصلها كالتالي:

١ـ الـ Selector او ما يمكننا تسمية العنصر المستهدف:

كما قمنا بالتوضيح سابقاً تحتاج الـ CSS لمعرفة من هو العنصر المستهدف الذي ستقوم بتطبيق التنسيقات المرغوبة عليه وسنذكر أبرز ثلاث طرق لعمل ذلك:



الطريقة الأولى: Element Selector

في هذه الطريقة تقوم بإستهداف عنصر بإستخدام الوسم ذاته كما بالصورة السابقة قمنا بإستهداف جميع عناصر الوسم <p>.

لنأخذ مثال:

```
<!--HTML CODE-->
<p>Hello world!</p>
<p>One Million Arab Coder</p>
```

```
/*CSS CODE*/
p{
   color:red;
}
```

قم بتطبيق المثال السابق.

النتيجة كالتالي :

ـ لاحظ انه تم إستهداف جميع عناصر الوسم p دون أي إستثناء

ـ لاحظ عند تعريفك للـ Element Selector في الـ CSS تقوم بكتابة إسم الوسم بدون أي إضافات أخرى

الطريقة الثانية: ID Selector

قمنا بشرح فكرة الـ ID في وكما لاحظت أننا نقوم بتعريفة من أجل إستهداف عنصر واحد فريد من نوعة في مستند الـ HTML و يعتبر واحد من أهم الـ Selectors وأكثرها إستخداماً.

لنأخذ مثال:

```
<!--HTML CODE-->
<p id="paragraph">Hello world!</p>
<p>One Million Arab Coder</p>
```

```
/*CSS CODE*/
#paragraph{
   color:red;
}
```

قم بتطبيق المثال السابق.

النتيجة كالتالي :

ـ لاحظ انه تم إستهداف العنصر الذي يحمل الـ id المحدد فقط دون سواه.

ـ لاحظ عند تعريفك للـ id Selector في الـ CSS تقوم بكتابة # ثم تكتب إسم الـ id كما في مثالنا أعلاة #paragraph.

الطريقة الثالثة: Class Selector

قمنا بشرح فكرة الـ Class في وكما لاحظت أننا نقوم بتعريفة من أجل إستهداف عنصر او مجموعة من العناصر في مستند الـ HTML و يعتبر واحد من أهم الـ Selectors وأكثرها إستخداماً.

لنأخذ مثال:

```
<!--HTML CODE-->
<p class="paragraph">Hello world!</p>
<p class="paragraph">One Million Arab Coder</p>
<p>I don't have a class</p>
```

```
/*CSS CODE*/
.paragraph{
   color:red;
}
```

قم بتطبيق المثال السابق.

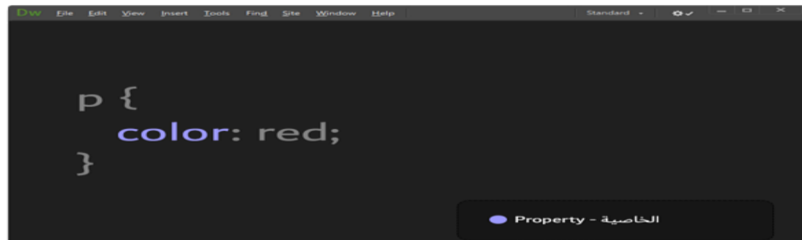
النتيجة كالتالي :

ـ لاحظ انه تم إستهداف عنصرين يملكان نفس إسم الكلاس paragraph

ـ لاحظ عند تعريفك للـ Class Selector في الـ CSS تقوم بكتابة . ثم تكتب إسم الـ class كما في مثالنا أعلاة .paragraph

٢ـ الـ Porperty او الخاصية:

الخطوة التالية بعد أن تقوم بتحديد العنصر المستهدف بإستخدام أحد الطرق أعلاة هو أن تبدأ بإضافة التنسيقات له، وللقيام بذلك يجب أن تقوم بتحديد نوع الخاصية المراد تطبيقها.



الخاصية في الصورة أعلاه color هي الخاصية المسؤولة عن تغيير لون الخط.

أنت غير ملزم بحفظ هذه الخصائص فسوف تنساها وانما مع الوقت والممارسة ستحفظها ودائماً بإمكانك الرجوع للمرجع الخاص بها لتعلمها ومعرفة فائدتها وطريقة إستخدامها

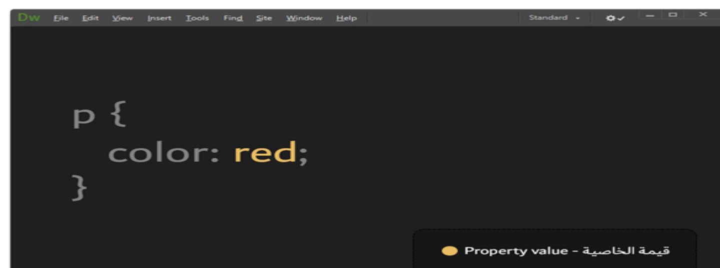
إضغط هنا للذهاب لمرجع خصائص الـ CSS26

https://www.w3schools.com/cssref/default.asp

ـ لاحظ طريقة كتابة الخاصية كالتالي color:

٢ـ الـ Property Value أو قيمة الخاصية :

الخطوة التالية بعد أن تقوم بتحديد العنصر المستهدف والخاصية المطلوب تنفيذها هي أن تقوم بتحديد قيمة هذه الخاصية.



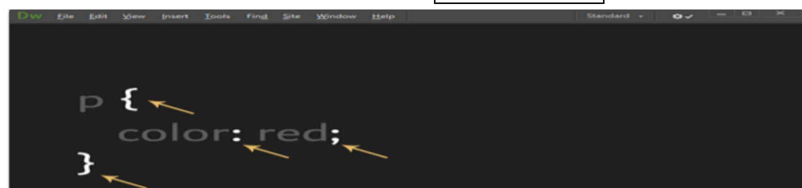
الصورة أعلاه توضح قيمة خاصية تغيير لون الخط حيث قمت بطلب تغيير لونة إلى اللون الأحمر red

ـ لاحظ هناك عدة طرق لتحديد اللون المطلوب بإمكانك تعلمها من خلال هذا المرجع13 وسنتطرق لها لاحقاً.

https://www.w3schools.com/cssref/pr_text_color.asp

ملاحظات مهمة:

- لاتنسى إغلاق كل خاصية بعد كتابة قيمتها بالفاصلة المنقوطة ؛ حتى وإن كانت الخاصية الأخيرة حتى تعتاد على ذلك وتتجنب الأخطاء.



هناك معايير محددة لعمل تسميات لأسماء الـ id و الـ class لجعل كودك مرتب ونظيف أحرص على التالي:

١ـ قم بإختيار أسماء منطقية حيث بمجرد مراجعتك لكود CSS تعرف اين هو العنصر المقصود بالتنسيق المحدد

على السبيل المثال: اذا لديك قسم في الموقع خاص بالعملاء قم بتسمية clients وليس sdfgd

```
<div id="clients"></div>
```

٢ـ اذا اردت كتابة أكثر من وصف لإسم القسم قم بفصلها بعلامة ـ

على سبيل المثال تريد عمل قسم لـ clients feedback لا تقوم بوضعة بالمسافة هكذا

```
<div id="clients feedback"></div>
```

فهذا الكود يعني أنك أعطيته أثنين كلاسات وليس واحد

قم بعمل التالي لجعلة كلاس واحد:

```
<div id="clients-feedback"></div>
```

٣ـ قم بكتابة أسماء الـ class والـ id بأحرف صغيرة

على سبيل المثال لا تقم بالتالي:

```
<div id="Clients"></div>
```

بل قم بالتالي:

```
<div id="clients"></div>
```

في الختام تذكر أن الـ CSS من أجمل مميزاتها أنك تقوم بكتابة الكود مرة واحدة وتطبيقة على أكثر من عنصر فلا تتجاهل هذا الميزة وتقوم بتكرار اكوادك لنفترض أنك أردت تطبيق نفس التنسيق على كلاسين مختلفين فلا تقم بالتالي:

```
.paragraph{
   color:red;
   }
.client-name{
   color:red;
   }
```

لاحظ نريد تطبيق نفس التنسيق بنفس اللون فبإمكاننا دمج هذا الكود بالشكل التالي:

```
.paragraph, .client-name{
  color:red;
 }
```

فقط قم بفصل كل كلاس عن الآخر بفاصلة ودعهم يتشاركوا نفس الخصائص

خاصية margin

- تذكر دائماً أن خاصية margin هي المسؤولة عن تحديد الهوامش الخارجية للعنصر كما هو موضح بالصورة.



نلاحظ في الصورة أن هذه الخاصية تتجزأ إلى ٤ أجزاء

- هامش من أعلى
- هامش من اليمين
- هامش من الأسفل
- هامش من اليسار

لنأخذ مثال:

```
<div class="margin-test"></div>
.margin-test{
  width: 980px;
  height: 100px;
  background-color: #0c214e;
  margin-top: 60px;
  margin-left: 80px;
}
```

النتيجة :



شاهد كيف قمت بإعطاء الصندوق هامش من اليسار ب ٨٠ بيكسل ومن الأعلى ٦٠ بيكسل.

كيف أستفيد من هذه الخاصية لتوسيط أي عنصر

خاصية margin تكون قيمتها الإفتراضية ٠ في حال عدم تغييرك لها و بإمكانك إعطائها قيمة auto لنأخذ هذا المثال:

```
.margin-test{
  margin-top: 60px;
  margin-left: auto;
}
```

عندما يشاهد المتصفح الكود أعلاه فسيقوم بالتالي:

١ـ إعطاء العنصر هامش من أعلى بمقدار ٦٠ بيكسل

```
margin-top: 60px;
```

٢ـ سيقوم بحساب الهامش تلقائياً إعتماداً على قيمة الهامش الذي قمنا بتحديدة للعنصر من جهة اليمين

```
margin-left: auto;
```

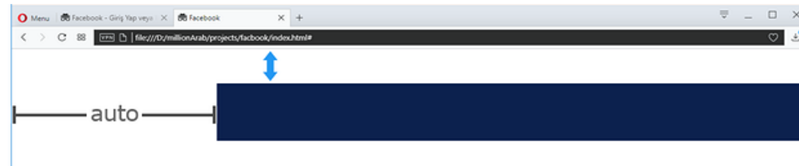
٣ـ سيقوم باعطاء العنصر هامش من اليمين بقيمة ٠ وذلك لانها القيمة الإفتراضية ولم نقم نحن بتغييرها

```
margin-right: 0;
```

٣ـ سيقوم باعطاء العنصر هامش من الأسفل بقيمة ٠ لنفس السبب

```
margin-bottom: 0;
```

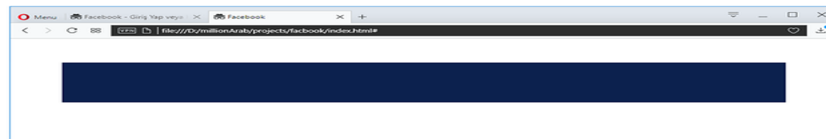
بما أننا حددنا للمتصفح بإعطاء العنصر هامش من اليمين(margin-right) بقيمة ٠ اذا سيقوم بإعطاء الأولوية لتطبيق هذا الأمر ولن يكون لـ auto نتيجة ملحوظة كالتالي:



كن ما الذي سيحصل اذا ما قمنا بإعطاء العنصر هامش من اليمين واليسار بنفس القيمة وهي auto 🤔؟

```
.margin-test{
  margin-top: 60px;
  margin-left: auto;
  margin-right: auto;
}
```

شاهد النتيجة :



قام المتصفح بإحتساب الهوامش من اليمين واليسار بشكل تلقائي مما جعل العنصر في وسط الصفحة.

هل الموضوع مربك لك ؟

بإمكانك تخيل الذي يحصل كشد الحبل، شاهد الصور التالية



يفترض بأنك الأن فهمت الفكرة، لننتقل خطوة للأمام ونقوم بكتابة الكود بطريقة أكثر إحترافا.

لماذا لا نقوم بدمج كل تلك الأكواد التي نحتاج لكتابتها لعمل margin في سطر واحد؟

سنقوم بذلك عبر تعميم إسم الخاصية على الجميع تحت مسمى margin

ركز على الصور التالية حتى تعرف كيف ستقوم بتحديد أي قيمة هي التي تخص الهامش الأعلى او الأسفل وغيرة



لاحظ قمت بكتابة كل القيم كالتالي :

```
margin:100px auto 100px auto;
```

قمنا بوضع كل القيم بجوار بعضها وسيقوم المتصفح بالمرور عليها وتقسيمها على الاتجاهات حسب إتجاة عقارب الساعة هكذا

أعلى يمين تحت يسار = ١٠٠ auto 100 auto

هل تذكر اننا سبق وقلنا بأننا يجب أن نحرص على عدم تكرار أكوادنا؟

هل تلاحظ اننا قمنا بتكرار القيم للإتجاهات المتضادة مثلا أعلى واسفل أعطيناها نفس القيمة ١٠٠ بيكسل بينما يمين ويسار أعطيناها نفس القيمة.

اذا هناك شي يربط بين هذه الاتجاهات فهناك الإتجاة وضدة فلنقوم بدمجهما مع بعض.



ليصبح لدينا الكود كالتالي

```
margin:100px auto;
```

وهنا طريقة أخرى بنفس الفكرة لكن هذه المرة قمنا بدمج هامش اليمين واليسار في قيمة واحدة




```
margin:100px auto 50px;
```

ماذا لو اردت إعطاء جميع الإتجاهات نفس القيمة ؟



حينها سنضع قيمة واحد وستحسب للجميع

```
margin:100px;
```

لتقوم بهذه التمارين حتى لا ترتبك

١ـ لنعطي العنصر هامش من اعلى ٢٠ بيكسل ومن اليمين ٣٠ بيكسل ومن اسفل ٥٠ بيكسل ومن اليسار ٢٥ بيسكل

```
margin:20px 30px 50px 25px;
```

٢ـ لنعطي العنصر من أعلى ٠ بيكسل ومن اليمين ٢٠ بيكسل ومن الأسفل ٠ بيكسل ومن اليسار ١٠ بيكسل

```
margin:0px 20px 0px 10px;
```

بإمكانك عدم كتابة px بعد القيمة ٠

```
margin:0 20px 0 10px;
```

٣ـ لنعطي العنصر من أعلى ٠ بيكسل ومن اليمين واليسار ٢٠ بيكسل ومن الأسفل ٣٠ بيكسل

```
margin:0px 20px 30px;
```

٤ـ لنعطي العنصر هامش ١٠ بيكسل من جميع الإتجاهات

```
margin:10px;
```

قم بكتابة الكود الذي سيعطي العنصر من أعلى ٠ بيكسل ومن اليمين auto ومن الأسفل ٠ بيكسل ومن اليسار auto

الجداول Table

نعرف جميعاً بأن أي جدول يحتوي على صفوف وكل صف يحتوي على خلايا كالتالي:

Cell - خلية | row - صف

Time	Sunday	Monday
12:00 pm	class 1	class 2
14:00 pm	class 3	class 4

table - جدول

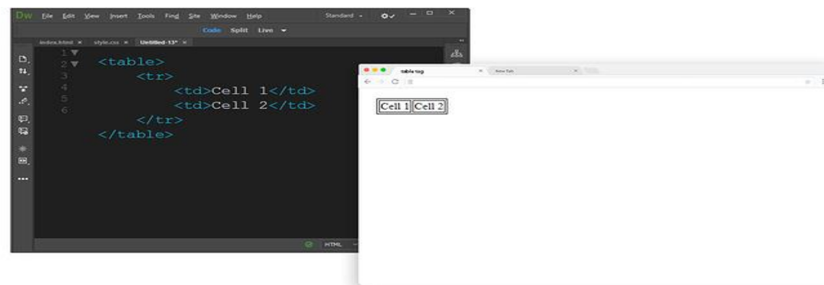
بنفس هذه الفكرة نستطيع إنشاء جداول في الـ HTML حيث يلزمنا التالي:

١ـ بدأية تعريف الجدول بإستخدام وسم <table>

٢ـ تعريف الصف بإستخدام الوسم <tr>

٣ـ تعريف خلايا الصف بإستخدام الوسم <td>

لنقم بعمل جدول بسيط:



لاحظ قمنا بتعريف الجدول بوسم table كالتالي:

```
<table>
    <!--تعريف الجدول-->
</table>
```

ثم قمت بعمل الصف الأول بوسم tr كالتالي:

```
<table>
```

```
   <tr>
<!--تعريف صف-->
   </tr>
</table>
```

ثم قمت بعمل كل خلية بداخل الجدول بالوسم td كالتالي:

```
<table border="1">
   <tr>
      <td>Cell 1</td> <!--الخلية الأولى في الصف-->
      <td>Cell 2</td> <!--الخلية الثانية في الصف-->
   </tr>
</table>
```

اذا ما قمنا به في الكود أعلاة هو تعريف جدول يحتوي على صف واحد وبداخل هذا الصف يوجد خليتين كما هو موضح في الصورة أعلاة.

لنقوم بعمل الجدول التالي:

Time	Sunday	Monday
12:00 pm	class 1	class 2
14:00 pm	class 3	class 4

نحتاج إلى أن نقوم بتعريف الجدول ومن ثم تعريف ثلاثة صفوف كالتالي:

```
<table border="1">
< tr ><!-- الصف الأول--></ tr >
< tr ><!-- الصف الثاني--></ tr >
< tr ><!-- الصف الثالث--></ tr >
</table>
```

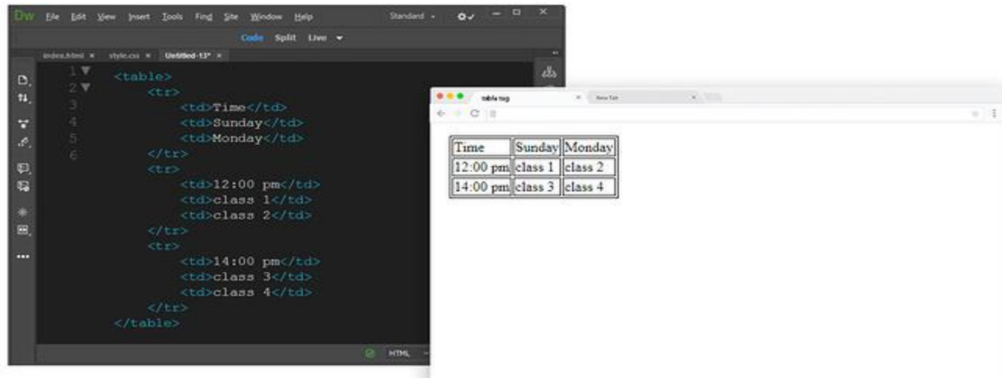
كما نلاحظ لدينا في كل صف ثلاث خلايا اذا لنقوم بإضافتها كالتالي:

```
<table border="1">
   <tr>
<td><!--الخلية الأولى بالصف الأول--></td>
<td><!--الخلية الثانية بالصف الأول--></td>
<td><!--الخلية الثالثة بالصف الأول--></td>
   </tr>
   <tr>
<td><!--الخلية الأولى بالصف الثاني--></td>
<td><!--الخلية الثانية بالصف الثاني--></td>
<td><!--الخلية الثالثة بالصف الثاني--></td>
   </tr>
   <tr>
<td><!--الخلية الأولى بالصف الثالث--></td>
<td><!--الخلية الثانية بالصف الثالث--></td>
<td><!--الخلية الثالثة بالصف الثالث--></td>
   </tr>
  </table>
```

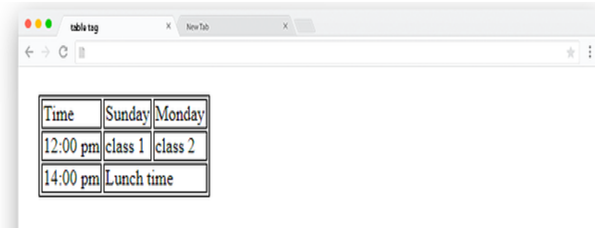
لنقم الأن بإضافة المحتوى الخاص بكل خلية كالتالي:

```
<table border="1">
   <tr>
     <td>Time</td>
     <td>Sunday</td>
     <td>Monday</td>
   </tr>
   <tr>
     <td>12:00 pm</td>
     <td>class 1</td>
     <td>class 2</td>
   </tr>
   <tr>
     <td>14:00 pm</td>
     <td>class 3</td>
     <td>class 4</td>
   </tr>
  </table>
```

النتيجة:



لنفترض أننا الآن نريد أن نقوم بعمل الجدول التالي:



لاحظ اننا الآن لدينا في الصف الثالث خليتين فقط بينما في بقية الصفوف لدينا ثلاث خلايا

Time	Sunday	Monday
12:00 pm	class 1	class 2
14:00 pm	Lunch time	

لنقم بعمل الجدول كما قمنا سابقاً ولكن هذه المرة الصف الثالث سيحتوي على خليتين فقط كالتالي:

```
<table border="1">
   <tr>
     <td>Time</td>
     <td>Sunday</td>
     <td>Monday</td>
   </tr>
   <tr>
     <td>12:00 pm</td>
     <td>class 1</td>
     <td>class 2</td>
   </tr>
   <tr>
     <td>14:00 pm</td>
     <td>Lunch time</td>
   </tr>
</table>
```

النتيجة:



هل تلاحظ اننا لم نحصل على النتيجة المطلوبة حيث وكأنه قد تم حجز مكان الخلية الثالثة في الصف الثالث بينما كنا نريد أن نقوم بدمج خليتين في خلية واحدة كالتالي:

Time	Sunday	Monday
12:00 pm	class 1	class 2
14:00 pm	Lunch time	

في الجداول نستطيع دمج الخلايا بطريقتين:

١ـ دمج الخلايا أفقياً colspan:

حيث يتم دمج خلية مع العدد المطلوب من الخلايا التي تليها كالتالي:

للقيام بهذا نستخدم colspan attribute **لنقم بتجربة ذلك على مثالنا السابق ولنرى اذا ما كنا سنحصل على النتيجة التي أردناها.**

للقيام بذلك سنحتاج إلى أن نقوم بطلب دمج خليتين بإستخدام "colspan="2 في وسم الخلية التي نرغب بتطبيق ذلك عليها وفي حالتنا نريد تطبيق ذلك على الخلية الثانية في الصف الثالث كالتالي:

```
<table border="1">
   <tr>
     <td>Time</td>
     <td>Sunday</td>
     <td>Monday</td>
   </tr>
   <tr>
     <td>12:00 pm</td>
     <td>class 1</td>
     <td>class 2</td>
   </tr>
   <tr>
     <td>14:00 pm</td>
     <td colspan="2">Lunch time</td>
   </tr>
</table>
```

النتيجة:

Time	Sunday	Monday
12:00 pm	class 1	class 2
14:00 pm	Lunch time	

رائع! حصلنا على الشكل المطلوب.

١ـ دمج الخلايا عمودياً rowspan:

حيث يتم دمج خلية مع العدد المطلوب من الخلايا التي تقع أسفلها كالتالي:

لنقوم بعمل الجدول التالي:

Time	Sunday	Monday
12:00 pm	class 1	class 2
	Lunch time	

في البدأية لنقم بحذف الخلية التي تحتوي على pm١٤:٠٠ ونشاهد النتيجة

```
<table border="1">
    <tr>
      <td>Time</td>
      <td>Sunday</td>
      <td>Monday</td>
    </tr>
    <tr>
      <td>12:00 pm</td>
      <td>class 1</td>
      <td>class 2</td>
    </tr>
    <tr>
      <td colspan="2">Lunch time</td>
    </tr>
</table>
```

النتيجة:

<table><tr><td>Time</td><td>Sunday</td><td>Monday</td></tr><tr><td>12:00 pm</td><td>class 1</td><td>class 2</td></tr><tr><td colspan="2">Lunch time</td></tr></table>

الأن لنقم بإضافة "rowspan="2 لوسم الخلية التي تحتوي على ١٢:٠٠pm ليتم دمجها كخليتين عمودياً

```
<table border="1">
    <tr>
      <td>Time</td>
      <td>Sunday</td>
      <td>Monday</td>
    </tr>
    <tr>
      <td rowspan="2">12:00 pm</td>
      <td>class 1</td>
      <td>class 2</td>
    </tr>
    <tr>
      <td colspan="2">Lunch time</td>
    </tr>
</table>
```

النتيجة:

<table><tr><td>Time</td><td>Sunday</td><td>Monday</td></tr><tr><td rowspan="2">12:00 pm</td><td>class 1</td><td>class 2</td></tr><tr><td colspan="2">Lunch time</td></tr></table>

رائع! حصلنا على الشكل المطلوب.

ملاحظة: تم وضع هذا الدرس من أجل توضيح ما سنحتاج إلية في درس تطبيقي ولم يتم وضعه لتغطية جميع خصائص الجداول لهذا أرجوا الإطلاع على هذا المصدر لمعرفة المزيد عن الجداول في الـ HTML:

https://www.w3schools.com/html/html_tables.asp